15.5A Bidirectional Current Sensor - ACS711 - Trēo™ Module

SModule Features

- Allegro ACS711
- 11mA Precision
- RoHS Compliant
- Software Library
- NightShade Trēo™ Compatible
- Breakout Headers

ACS711 Features

(from Allegro)

- High-Sensitivity Sensing Up to 15.5A (AC or DC)
- 100kHz Bandwidth (Response <550ns)
- 0.6mΩ Resistance Results in Low Losses

Applications

- Power Monitoring
- Robotics
- Battery Management
- Solar or Wind Power Stations

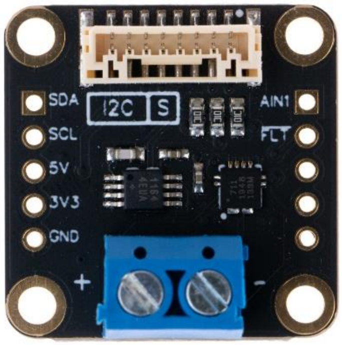
Trēo™ Compatibility

Electrical

Communication	I2C
Max Current, 3.3V	7mA
Max Current, 5V	1mA

Mechanical

- 25mm x 25mm Outline
- 20mm x 20mm Hole Pattern
- M2.5 Mounting Holes



Description

The ACS711 Trēo™ Module is a 15.5A Bidirectional Current Sensor module that that features Allegro's ACS70331 IC. It can sense ±15.5A of DC or AC current with a resolution of 11mA. This sensor has an electrical resistance of only 0.6mΩ, resulting in low losses. This module is a part of the NightShade Treo system, patent pending.

Table of Contents

1 Summary

The ACS711 module provides the measurement for a current path. Current flowing from plus to minus is positive. The ACS711 outputs an analog voltage proportional to the current flow which is measured with a MAX11644 ADC. The module is initialized with the begin() function and then measurements are taken with the read() function. The setOffset() function can be used to a calibrate the sensor zero.

2 What is Trēo™?

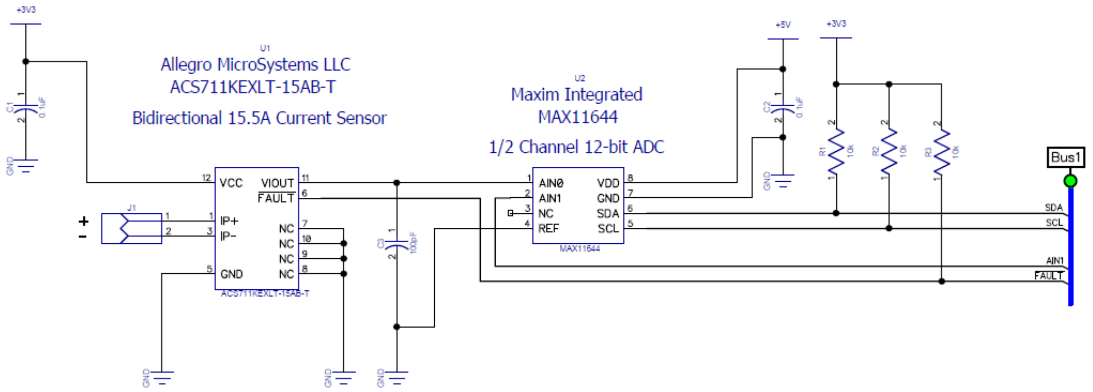
NightShade Trēo is a system of electronic modules that have standardized mechanical, electrical, and software interfaces. It provides you with a way to quickly develop electronic systems around microprocessor development boards. The grid attachment system, common connector/cabling, and extensive cross-platform software library allow you more time to focus on your application. Trēo is supported with detailed documentation and CAD models for each device.

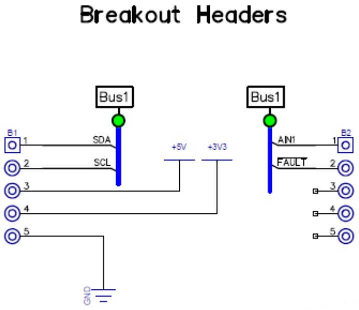
Learn more about Trēo here.

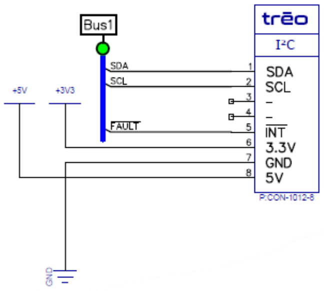
3 Electrical Characteristics

	Minimum	**Nominal**	**Maximum**
Voltages			
$V_{i/o}$ (SDA, SCL, INT)	-0.3V	-	3.6V
$V_{3.3V}$	3.1V	3.3V	3.5V
V_{5V}	4.8V	5.0V	5.2V
Measurement			
Bandwidth	-	-	100kHz
Range	-15.5A AC or DC	-	+15.5A AC or DC
Precision		11mA/LSB	
Error (25°C)	-7%	-	+7%
Slave Address		0x36	
Operating Temperature	-25°C	-	+85°C

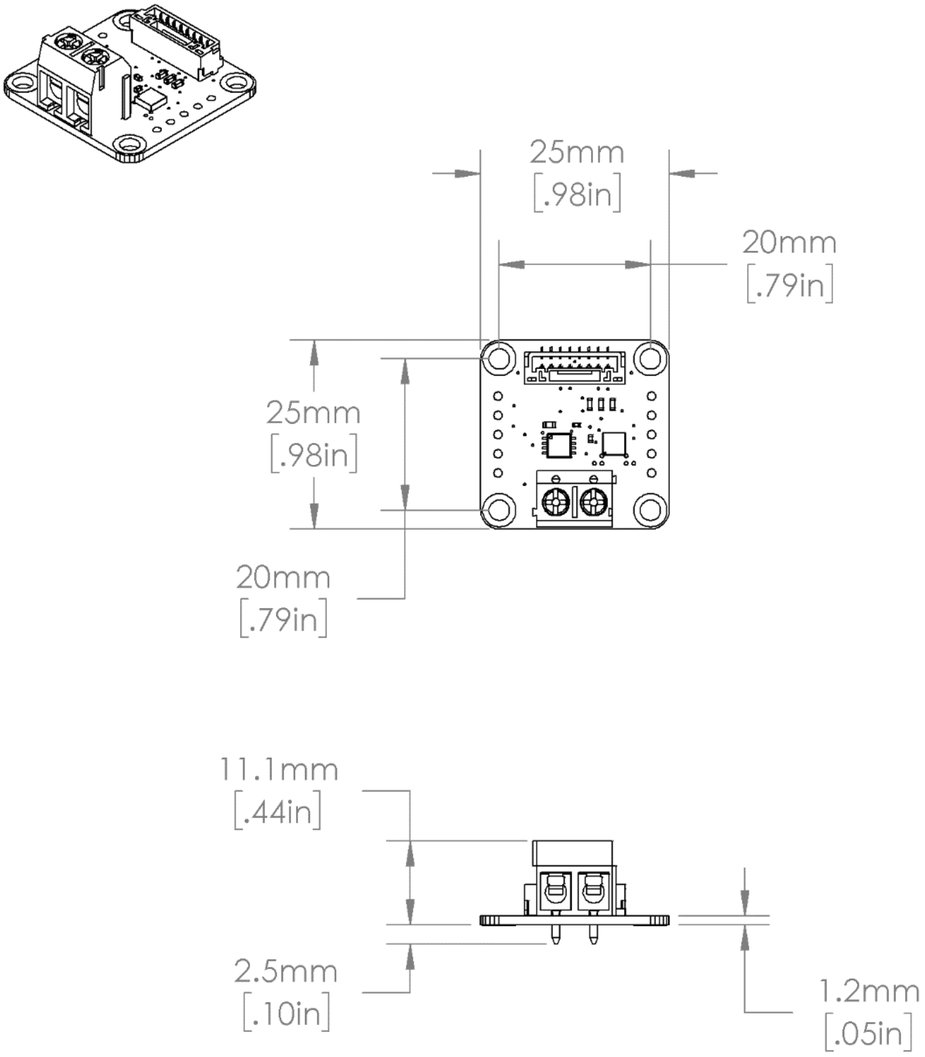
4 Electrical Schematic







5 Mechanical Outline



6 Example Arduino Program

```
/**********************************************************
  ACS711_15A_CurrentSensor - NightShade_Treo by NightShade Electronics

  This sketch demonstrates the functionality of the
  NightShade Trēo ACS711 15A current sensing module.
  (NSE-1131-1) It prints the current value passing
  through the sensor to Serial at 115200 baudrate.

  Created by Aaron D. Liebold
  on February 15, 2021

  Links:
  NightShade Trēo System: https://nightshade.net/treo
  Product Page: https://nightshade.net/product/treo-15a-current-sensor-acs711/

  Distributed under the MIT license
  Copyright (C) 2021 NightShade Electronics
  https://opensource.org/licenses/MIT
**********************************************************/

// Include NightShade Treo Library
#include <NightShade_Treo.h>

// Declare Objects
NightShade_Treo_ACS711 current(1);

void setup() {
  current.begin();
  Serial.begin(115200);
}

void loop() {
  Serial.print(current.read());
  Serial.println("mA");
  delay(500);
}
```

7 Library Overview (C++ & Python)

C++ Class

NightShade_Treo_ACS70331 <classObject>();

Python Module

<classObject> = NightShade_Treo.ACS70331();

7.1 Constructors

NightShade_Treo_ACS70331(int port, uint8_t slaveAddress, uint32_t clockSpeed)

Creates an ACS70331 object.

Arguments:

port	Integer of the I2C port used (e.g. 0 = “/dev/i2c_0”)
slaveAddress	7-bit slave address
clockSpeed	desired clock speed for the bus

Returns:

Nothing

NightShade_Treo_ACS70331(int port)

Creates a ACS70331 object assuming the default slave address and clock speed.

Arguments:

port	Integer of the I2C port used. (e.g. 0 = “/dev/i2c_0”)

Returns:

Nothing

7.2 Methods

begin()

Initializes the ACS70331 module.

Arguments:

None

Returns:

Error	0 = Success

read()

Returns a measurement from the current sensor.

Arguments

None

Returns

Returns the current in mA. (signed int)

setOffest(int offset)

Sets the zero-point offset, for calibration.

Arguments

offset	Offset value in mA.

Returns

Nothing